

## **IN THE CLAIMS**

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~striketrough~~. When striketrough cannot easily be perceived, or when five or fewer characters are deleted, ~~[[double brackets]]~~ are used to show the deletion. The status of each claim is indicated with one of (original), (currently amended), (cancelled), (withdrawn), (new), (previously presented), or (not entered). Please AMEND claims 1, 3-6, 8-11, and 13-15 in accordance with the following:

1. (CURRENTLY AMENDED) A method for controlling a plurality of threads that perform parallel processing, the method comprising:

monitoring a number of running threads performing parallel processing and a number of standby threads that are in a standby state for a predetermined time period; and

setting a necessary number of standby threads in accordance with the number of running threads during the predetermined time period;

comparing the number of the standby threads with ~~[[a]]~~ the necessary number of the ~~running~~ standby threads at a predetermined time interval; and

terminating ~~an amount~~ a number of ~~[[the]]~~ standby threads exceeding the necessary number when the number of ~~[[the]]~~ standby threads is greater than the necessary number.

2. (CANCELLED)

3. (CURRENTLY AMENDED) The method according to claim 1,  
wherein the necessary number includes a maximum number of ~~[[the]]~~ running threads during a predetermined time period, and

wherein said comparing includes comparing the maximum number of ~~[[the]]~~ running threads and the number of ~~[[the]]~~ standby threads.

4. (CURRENTLY AMENDED) The method according to claim 1,  
wherein the necessary number includes an average number of the number of ~~[[the]]~~ running threads during a predetermined time period, and

wherein said comparing includes comparing the average number of [[the]] running threads and the number of [[the]] standby threads.

5. (CURRENTLY AMENDED) The method according to claim 1,  
wherein the necessary number includes a product obtained by multiplying the number of [[the]] running threads during a predetermined time period by a predetermined coefficient, and  
wherein said comparing compares the product and the number of [[the]] standby threads.

6. (CURRENTLY AMENDED) A controller for controlling a plurality of threads that perform parallel processing, the controller comprising:

a thread management table storing thread information of the plurality of threads, wherein the thread information includes a number of running threads performing parallel processing and a number of standby threads that are in a standby state;

a thread management circuit requesting thread generation based on the number of [[the]] standby threads stored in the thread management table, and requesting a standby thread to run;

a comparison circuit setting a necessary number of standby threads in accordance with the number of running threads included in the thread information, and comparing the number of [[the]] standby threads with [[a]] the necessary number at a predetermined time interval; and

a termination circuit terminating ~~an amount~~ a number of [[the]] standby threads exceeding the necessary number when the number of [[the]] standby threads is greater than the necessary number.

7. (CANCELLED)

8. (CURRENTLY AMENDED) The controller according to claim 6, wherein the necessary number is a maximum value of [[the]] running threads during a predetermined time period.

9. (CURRENTLY AMENDED) The controller according to claim 6, wherein the necessary number is an average value of [[the]] running threads during a predetermined time period.

10. (CURRENTLY AMENDED) The controller according to claim 6, wherein the necessary number is a product obtained by multiplying the number of [[the]] running threads during a predetermined time period by a predetermined coefficient.

11. (CURRENTLY AMENDED) A computer readable storage medium storing a program for controlling at least one processor to execute a plurality of threads that perform parallel processing, according to a method comprising:

monitoring a number of running threads performing parallel processing and a number of standby threads that are in a standby state for a predetermined time period; and  
setting a necessary number of standby threads in accordance with the number of running threads during the predetermined time period;

comparing the number of [[the]] standby threads with [[a]] the necessary number of the ~~running standby threads at a predetermined time interval;~~ and

terminating an amount of [[the]] standby threads exceeding the necessary number when the number of [[the]] standby threads is greater than the necessary number.

12. (CANCELLED)

13. (CURRENTLY AMENDED) The storage medium according to claim 11,  
wherein the necessary number refers to a maximum number of [[the]] running threads during a predetermined time period, and

wherein said comparing includes comparing the maximum number of [[the]] running threads and the number of [[the]] standby threads.

14. (CURRENTLY AMENDED) The storage medium according to claim 11,  
wherein the necessary number refers to an average number of the number of [[the]] running threads during a predetermined time period, and

wherein said comparing includes comparing the average number of [[the]] running threads and the number of [[the]] standby threads.

15. (CURRENTLY AMENDED) The storage medium according to claim 11,  
wherein the necessary number includes a product obtained by multiplying the number of [[the]] running threads during a predetermined time period by a predetermined coefficient, and  
wherein said comparing compares the product and the number of [[the]] standby threads.